

Micro-talleres Arduino: una experiencia piloto en informática para ingenierías en el aprendizaje de la programación

Jorge Sales, Gabriel Recatalá, José V. Martí
Departamento de Ingeniería y Ciencia de los Computadores
Universitat Jaume I
Castellón
{salesj, grecata, vmarti}@uji.es

Resumen

En este trabajo se presenta una propuesta de mejora metodológica llevada a cabo durante el curso 14/15 en una asignatura de introducción a la programación de ordenadores en el ámbito de las ingenierías. Esta asignatura debe abordar, entre otros contenidos, la programación de computadores, una de las competencias de mayor dificultad para los estudiantes. Esta mejora consiste en la implantación de micro-talleres de programación en grupo utilizando la plataforma de hardware libre Arduino, en el contexto de las clases de problemas de la asignatura. El objetivo de estos talleres es favorecer la comprensión y el aprendizaje por parte de los estudiantes de dos conceptos elementales en los que venimos encontrando serias dificultades de aprendizaje: estructuras repetitivas (bucles) y condicionales. Las encuestas pasadas a los alumnos revelan una valoración muy positiva, lo que demuestra la efectividad de esta mejora metodológica, apuntando a un algo grado de motivación y a la consecución de los objetivos que nos habíamos planteado.

Abstract

This paper presents a proposal for methodological improvement made during the course year 14/15 in an introductory computer programming course in the field of engineering degrees. This course addresses, among other contents, computer programming, one of the most difficult competences for students. This improvement involves the implantation of group programming micro-workshops using the open hardware platform Arduino. The goal of these workshops is to improve the understanding and learning of two elementary concepts, in which we have been encountering severe learning difficulties: loops and conditionals. The effectiveness of this methodological improvement, measured primarily through surveys, indicate a very positive evaluation by the students, pointing to a high degree of motivation and achievement of the initial objectives.

Palabras clave

Arduino, Micro-talleres, Programación, Trabajo en equipo, Participación activa.

1. Introducción

Los actuales planes de estudio de las ingenierías de ámbito industrial cuentan con la asignatura Informática en el bloque de materias básicas. La asignatura es de primer curso y primer semestre y se estructura en sesiones de teoría (clase magistral tradicional), sesiones de problemas (en grupos reducidos, de hasta 30 estudiantes, donde se plantean y resuelven problemas utilizando metodologías de aprendizaje basado en problemas) y sesiones de laboratorio (de hasta 25 estudiantes, donde los estudiantes resuelven problemas, en grupos de 2-3 personas, mediante algoritmos utilizando el ordenador). Esta asignatura se implantó en el curso académico 2010-11, año en el que se empezaron a implantar los nuevos grados de ingenierías adaptados al EEES. Desde ese momento, los profesores de la asignatura hemos estado introduciendo continuas innovaciones docentes tanto en las sesiones de laboratorio como de problemas [1, 2, 3].

De la experiencia del profesorado en los últimos años desde su implantación, se observa que una de las partes en la que los alumnos encuentran más dificultades es en el aprendizaje y dominio de la programación. Según algunos autores [4, 5], dicha dificultad se ve agravada por aspectos como la actitud de rechazo de los estudiantes ante una asignatura que consideran ajena, o la heterogeneidad de la base previa de conocimientos de los mismos.

2. Propuesta de mejora: micro-talleres Arduino

En este curso hemos implantado la realización de micro-talleres de programación utilizando la platafor-

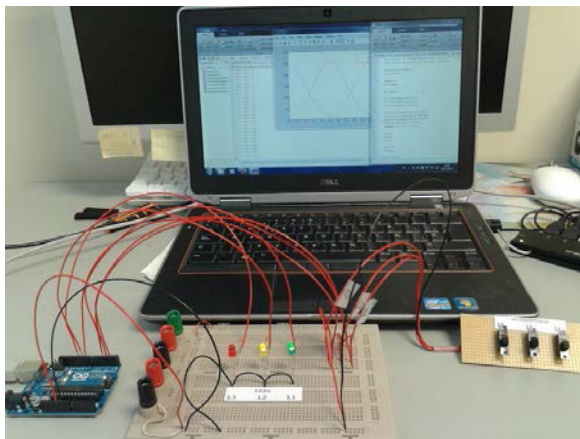


Figura 1: Vista de la plataforma Arduino propuesta: Ordenador portátil ejecutando Matlab, panel de interruptores, placa de pruebas con LEDs y placa Arduino.

ma de hardware libre Arduino (ver Figura 1). Esta plataforma consiste en una placa con un microcontrolador y diversos puertos de entrada/salida. Existe una librería para poder programarla en Matlab, el lenguaje que se estudia en la asignatura. Les denominamos micro-talleres porque tienen una duración máxima de 10 minutos. Se realizan en grupos de 4-5 estudiantes y de manera simultánea durante desarrollo tradicional las sesiones de problemas de la asignatura (ver Figura 2). Con la ayuda de un segundo profesor de apoyo, los diferentes grupos “aparcen” temporalmente la sesión de resolución de problemas y van “pasando” por el taller para desarrollar un pequeño programa. Durante su programación lo van probando sobre la plataforma real hasta conseguir el resultado deseado.

El objetivo de los talleres es favorecer la comprensión y el aprendizaje por parte de los estudiantes de dos conceptos elementales: bucles y condicionales. De nuestra experiencia en la asignatura, observamos que los alumnos encuentran muchas dificultades en los siguientes aspectos: (1) entender el concepto de bucle, (2) saber elegir el tipo de estructura repetitiva más adecuada en cada caso (e.g. bucle for o bucle while), (3) entender las estructuras condicionales y (4) distinguir entre estructuras condicionales y repetitivas. Los talleres propuestos están precisamente diseñados para practicar estos conceptos de una manera visual y práctica. Al mismo tiempo, el uso de estas plataformas hardware acerca a los estudiantes a ejemplos reales de aplicación directamente relacionados con sus titulaciones.

En este primer curso de implantación se han desarrollado 4 talleres, que consisten en la realización de secuencias luminosas utilizando LEDs (diodos emisores de luz) como salida (contador binario, semáforo, patrones varios) y también en función del valor de unos interruptores utilizados como entrada. Toda la documenta-

Parte I: Variables, operadores, gráficos, etc.

- S1: Boletín 1. Calculadora escalar
- S2: Boletín 2. Calculadora vectorial y matricial
- S3: Boletín 3: Gráficos y Ficheros-M
- S4: Boletín 4: Funciones
- S5: Control de Problemas (Boletines 1-4)

Parte II: Bucles y Condicionales

- | | | |
|---|-----------------|------------------|
| S6: Boletín 5: Condicionales (I) | T1: Cond. (I) | Talleres Arduino |
| S7: Boletín 6: Bucles (I) | T2: Bucles (I) | |
| S8: Boletín 5: Condicionales (II) | T3: Cond. (II) | |
| S9: Boletín 6: Bucles (II) | T4: Bucles (II) | |
| S10: Control de Problemas (Boletines 5-6) | | |

Figura 2: Estructura de las sesiones de problemas: 10 sesiones divididas en dos partes. Se puede observar cómo se han introducido los Talleres Arduino en la Parte II relativa a Bucles y Condicionales.

ción necesaria para la realización del taller (boletines, vídeos, imágenes explicativas, etc.) se encuentra disponible on-line en un sitio web¹ de libre acceso tanto para los estudiantes como para la comunidad educativa en general.

El hecho de romper con la monotonía de la clase y realizar un ejercicio práctico que permite ser probado durante la realización del mismo en un hardware real aporta un valor pedagógico a la enseñanza y una motivación extra a los alumnos a la hora de entender los conceptos que se pretende que aprendan. La experiencia ha sido valorada muy positivamente por los estudiantes, mejorando tanto su motivación como la comprensión de los conceptos de programación que procuramos abordar, y así lo evidencian tanto sus comentarios como los resultados de las encuestas que les hemos realizado en algunos de los talleres.

3. Estado del Arte

La enseñanza de la programación y de las habilidades relacionadas ya desde edades tempranas es cada vez más importante [6]. En la actualidad, los dispositivos móviles, las fuentes de consulta de Internet y las redes sociales son de uso común entre los alumnos, tal como se apunta en [7]. Este hecho favorece, sin duda, la familiaridad con la tecnología y proporciona, por tanto, un punto de acceso para proponer mé-

¹<https://sites.google.com/a/uji.es/arduino/matlab>

todos de aprendizaje aprovechando esta circunstancia. Algunas actividades típicas consisten en proponer a los alumnos que resuelvan problemas interactivos con sus dispositivos móviles, como en [8] o en proporcionarles el material necesario para construir un robot desde cero, e incluso organizar pequeñas competiciones para ejercitar las habilidades adquiridas, como en [9]. Con este punto de partida y con el propósito de estimular el interés, en algunos casos se propone a alumnos de primeros cursos la realización de trabajos con entornos de desarrollo amigables como por ejemplo Arduino, tanto como parte de las prácticas regladas (como en [10]) como constituyendo proyectos en los que los alumnos persiguen un determinado objetivo, como en [11]. Estos dos últimos trabajos nos han servido de inspiración para mejorar la metodología de enseñanza-aprendizaje en nuestra asignatura, utilizando en nuestro caso una plataforma de bajo coste basada en Arduino y con un tiempo de preparación muy razonable.

4. Descripción de la plataforma Arduino utilizada

El montaje utilizado en los talleres de Arduino se basa en el ejemplo de pulsador para encender un LED que se encuentra en la web de Arduino², en la sección de tutoriales. Adicionalmente, se pueden encontrar más detalles acerca de su montaje en la propia web de los talleres. La plataforma de desarrollo que hemos utilizado consta de los siguientes elementos:

- Placa Arduino UNO³. Se trata de una plataforma de hardware libre, basada en una placa con un microcontrolador (ATmega328) y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.
- Placa de circuito perforada de prototipado con tres interruptores. Se utiliza para generar las señales de entrada a la placa Arduino.
- Placa de pruebas (protoboard) donde se ubican los LEDs y se realizan las conexiones.
- Cable de conexión USB (tipo A-B).
- Ordenador portátil con Matlab. Se utiliza la librería MATLAB Support Package for Arduino⁴, disponible de forma gratuita en internet. Esta librería permite utilizar MATLAB para comunicarse con la placa Arduino con un cable USB.

El coste de todo el material necesario para poner en marcha la plataforma, sin incluir el coste del ordenador portátil y licencias de software, es de unos 50€. El tiempo necesario para su montaje es de unas 5h.

²<http://arduino.cc/en/tutorial/button>

³<http://arduino.cc/en/main/arduinoBoardUno>

⁴<http://es.mathworks.com/hardware-support/arduino-matlab.html>



Figura 3: Alumnos realizando el taller de Arduino.

5. Descripción de los talleres

En todos los talleres, a los alumnos se les proporciona un pequeño boletín (disponible en la web) y un esqueleto del programa a modificar, ya preparado en el ordenador, que contiene las siguientes partes:

- El código de inicialización de la tarjeta y de las comunicaciones entre Matlab y la tarjeta a través del puerto USB.
- La declaración de las variables necesarias. La inicialización se proporciona de manera parcial, dependiendo del boletín y del problema en concreto.
- La estructura del algoritmo en pseudocódigo comentado.

A partir de este momento, el profesor de apoyo encargado del taller, explica brevemente a los alumnos el objetivo que han de conseguir y les va guiando en la realización de cada paso, asegurándose de que alcanzan los objetivos didácticos perseguidos (ver Figura 3).

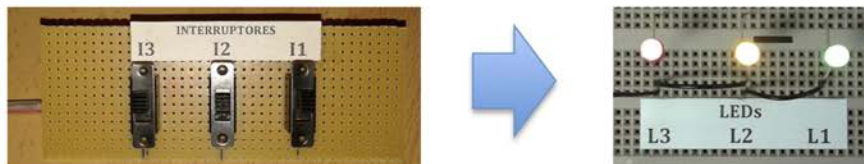
A continuación, se describe con detalle en qué consiste cada taller y cuales son los objetivos didácticos que se pretenden alcanzar.

5.1. Micro-taller 1: Condicionales (I)

Este taller lo hemos realizado dentro de la sesión 6 de problemas (ver Figura 2). En esta sesión, y siguiendo la metodología tradicional en la asignatura, los alumnos, organizados en pequeños grupos de trabajo, resuelven problemas que requieren el uso de operadores lógicos y relacionales, así como estructuras condicionales. Para ello se les hace entrega de un boletín de problemas (Boletín 5) que contiene tanto problemas para realizar en clase como problemas para realizar en casa y que posteriormente entregan a través del Aula Virtual (la plataforma Moodle en nuestra universidad) en unas fechas prefijadas. Este boletín se utiliza tanto

Micro-Taller Arduino 1: Condicionales (I)

Ejercicio 1: Crea un programa que permita encender cada uno de los LEDs con un interruptor.



Ayuda:

Para iniciar la comunicación con Arduino has de crear el siguiente objeto **a**:

```
>> a = arduino('COM3');
```

Para leer el valor de los interruptores puedes utilizar esta función:

```
V11 = digitalRead(a, I1); % Lee el valor de I1 en V11
```

Para encender o apagar un LED puedes utilizar esta función:

```
digitalWrite(a, L1, 1); % Enciende L1
```

```
digitalWrite(a, L1, 0); % Apaga L1
```

Para resolver este ejercicio tienes que leer el valor de cada uno de los interruptores y decidir si has encendido o apagado el LED correspondiente utilizando un condicional:

```
Si I1 está accionado
    Enciende L1
Sino
    Apaga L1
```

Sugerencia 1:

Es útil mostrar por la consola de Arduino el estado de los interruptores. Puedes hacerlo con la función **disp**:

```
disp([num2str(VI3) ' - ' ... ]);
```

Sugerencia 2:

Si has resuelto el problema como se ha indicado antes, observarás que es necesario ejecutarlo cada vez que quieras comprobar el nuevo estado de los interruptores y cambiar el estado de los LEDs. Para que el programa se repita indefinidamente, puedes situar los condicionales dentro de un bucle infinito como este:

```
while true
    Condicionales
    pause(0.100); % Esperar 100ms
end
```

Figura 4: Enunciado del micro-taller Arduino utilizado en la sesión 6 de problemas. Se trata del primero de los cuatro talleres Arduino en el que los alumnos entran en contacto con la plataforma y deben resolver un problema utilizando Condicionales.

para la sesión 6 de problemas (Condicionales parte I) como para la sesión 8 (Condicionales parte II).

El objetivo del micro-taller 1 de Arduino es que los alumnos entiendan mejor y de una forma práctica el concepto de condicional. Siguiendo la metodología ya explicada anteriormente, los distintos grupos de alumnos que se encuentran resolviendo problemas van pasando por el taller. En este primer taller, los alumnos tienen una primera toma de contacto con la plataforma desarrollada (interruptores de entrada y LEDs de salida) y el entorno de programación. El ejercicio a resolver consiste en crear un programa que permita encender cada uno de los LEDs con un interruptor. El primer interruptor I1 encenderá el LED L1, el segundo interruptor I2 encenderá el LED L2, y el tercer interruptor I3 encenderá el LED L3.

A modo de guía, les proporcionamos el boletín que se muestra en la Figura 4, junto al esqueleto del código fuente que han de completar para realizar el ejercicio. En este boletín se explica el problema que han de resolver, la parte de inicialización de la tarjeta, variables a utilizar, estrategia (en pseudocódigo) para resolver el ejercicio, y uso de funciones de salida para mostrar resultados por pantalla.

Para poder realizar el taller en el tiempo previsto (entre 8-10 minutos), les proporcionamos el entorno de programación ya en funcionamiento y un esqueleto del programa ya realizado, de forma que sólo se tienen que preocupar en la parte de implementación y prueba.

Para resolver el problema planteado (encender cada uno de los LED según el estado de cada interruptor) deben hacer una estructura condicional de tipo if/else para cada par interruptor-LED. Cada paso que realizan en la programación, lo pueden ir probando en la tarjeta con solo pulsar el botón de ejecutar en el entorno Matlab. Hemos observado que esto les motiva mucho a la hora de realizar el programa y probar cambios que se les van ocurriendo sobre la marcha, perdiendo de esta forma el “miedo” a la programación.

Durante la programación les mostramos también el uso de la función `disp` para mostrar el resultado en pantalla. De esta forma, practican también la combinación de cadenas de texto y múltiples variables, un aspecto que normalmente les genera confusión.

Cuando han conseguido implementar correctamente las tres estructuras condicionales, tienen la ocasión de comprobar que el programa no funciona como ellos esperan, ya que sin el uso de una estructura repetitiva, el circuito sólo responde una vez en cada ejecución. En este momento, de gran valor didáctico, aprovechamos para explicarles que necesitan una estructura repetitiva (que se verá con más detalle en la próxima sesión) para que el programa no deje de ejecutarse y la estructura condicional se evalúe de manera indefinida (o el número de veces que ellos decidan). La manera más sencilla

	L3	L2	L1
0	0	0	1
1	0	1	0
2	1	0	0
3	0	1	0
4	0	0	1
5	0	1	0
6	1	0	0
...

Figura 5: Secuencia infinita que han de seguir los LEDs en el ejercicio planteado en el segundo taller Arduino.

lla de conseguir el resultado deseado es utilizando un bucle infinito. Una vez lo han implementado, pueden comprobar el correcto funcionamiento de los interruptores y los LEDs.

5.2. Micro-taller 2: Bucles (I)

Este taller se realiza dentro de la sesión 7 de problemas. Al igual que en la sesión anterior, los alumnos organizados en pequeños grupos de trabajo, resuelven problemas que, en este caso, requieren el uso de operadores lógicos y relacionales, así como estructuras repetitivas (bucles). Para ello se les hace entrega de un boletín de problemas (Boletín 6), que se utiliza tanto para la sesión 7 (Bucles parte I) como para la 9 (Bucles parte II).

El objetivo del micro-taller 2 es que los alumnos entiendan mejor y de una forma práctica el concepto de bucle. En este punto, los alumnos ya conocen la plataforma y entienden cómo funciona. El ejercicio a resolver consiste en crear un programa que permita encender cada uno de los LEDs utilizando una secuencia infinita predefinida (ver Figura 5). Como guía para la realización del taller, les hacemos entrega de un boletín que contiene los pasos a seguir. En la web de los talleres, además de este boletín, se encuentran unos vídeos con el resultado que se pretende conseguir, con el objetivo de facilitar su comprensión y aumentar también la motivación de los estudiantes. Además del boletín, disponen del esqueleto del código fuente que han de completar para realizar el ejercicio.

Como primera aproximación al problema, utilizarán 3 vectores (uno para cada LED) conteniendo la secuencia pedida. Puesto que la secuencia se repite cada 4 intervalos de tiempo, se puede resolver utilizando vectores de 4 elementos. Para recorrer los vectores utilizarán un bucle con una variable de 1 a 4 que se utilizará para hacer los accesos al vector.

Hemos observado que el tema de accesos a vectores suele resultar dificultoso para los estudiantes, especialmente mediante el uso de bucles, por lo que este ejercicio tiene también gran interés pedagógico a la hora de

practicar este aspecto de la programación. Durante la programación, al igual que en el caso anterior, pueden ir probando cómo se comporta el programa viendo los LEDs de la tarjeta. Además, les recordamos también el uso de la función `disp` para mostrar el resultado en pantalla.

Cuando han conseguido implementar correctamente el bucle, de forma similar al taller 1, tienen la ocasión de comprobar que el programa no funciona como ellos esperan, ya que sin el uso de una segunda estructura repetitiva la secuencia sólo se visualiza una vez en cada ejecución. En este momento el profesor aprovecha para introducir el concepto de bucle anidado. Siguiendo el paralelismo con el taller anterior, los estudiantes entienden fácilmente la necesidad de un bucle infinito que engloba al bucle que acaban de escribir.

A continuación, el profesor les plantea una ligera modificación, que consiste en utilizar vectores de 3 elementos en lugar de 4. Para conseguirlo, deben hacer que la variable que se utiliza en las iteraciones, por ejemplo `i`, tome los valores `i=1:3,2` en lugar de `i=1:4`. De esta manera, aprenden que la cuenta para el acceso a los vectores puede ser tanto creciente como decreciente.

Como propuesta para pensar en casa, les proponemos una ampliación que consiste en hacer la secuencia de encendido de los LEDs mediante transiciones suaves. Los detalles de esta propuesta y un vídeo demostrativo se encuentra en la web de los talleres.

5.3. Micro-taller 3: Condicionales (II)

El tercer taller tiene como objetivo volver al tema de los condicionales para afianzar conocimientos, y se realiza dentro de la sesión 8 de problemas. El problema a resolver consiste en utilizar los interruptores de entrada de forma que generando combinaciones entre ellos permita el encendido de los distintos LEDs que en esta ocasión son de colores. Hemos elegido el rojo, el naranja y el verde para simular un semáforo y facilitar la comprensión del enunciado. Les proporcionamos una tabla con el resultado que debe generar cada combinación. El objetivo pedagógico es que practiquen el uso de condicionales con múltiples condiciones, donde cada condición es una condición compuesta con operadores aritméticos y lógicos. En este punto es interesante recordarles que el caso por defecto (el último `else`) se suele utilizar para ejecutar una acción a tomar en el caso en que no se cumpla ninguna de las condiciones anteriores, que en el ejercicio sería apagar todos los LEDs.

Al igual que en los talleres anteriores, para conseguir un funcionamiento continuo del programa, los alumnos se dan cuenta de que necesitan una estructura repetitiva (un bucle infinito) que englobe la estructura condicional que acaban de escribir.

5.4. Micro-taller 4: Bucles (II)

El cuarto y último taller tiene como objetivo volver al tema de los bucles, tanto simples como anidados, y se realiza dentro de la sesión 9 de problemas. Continuando con la idea de las secuencias de iluminación del taller 2, en este caso se ha de implementar una cuenta binaria. Puesto que se dispone de 3 LEDs, se plantea como posible estrategia de resolución el uso de 3 vectores de 8 elementos, para poder alojar todas las combinaciones posibles. La solución propuesta en este caso es un bucle que recorre los vectores para `i=1:8`.

En este punto, consideramos de interés pedagógico comentar la escalabilidad de la solución al problema. Les planteamos cómo podrían solucionar el mismo problema para el caso de 4 o más LEDs. Se trata de que los alumnos propongan soluciones alternativas como por ejemplo el uso de bucles anidados, etc.

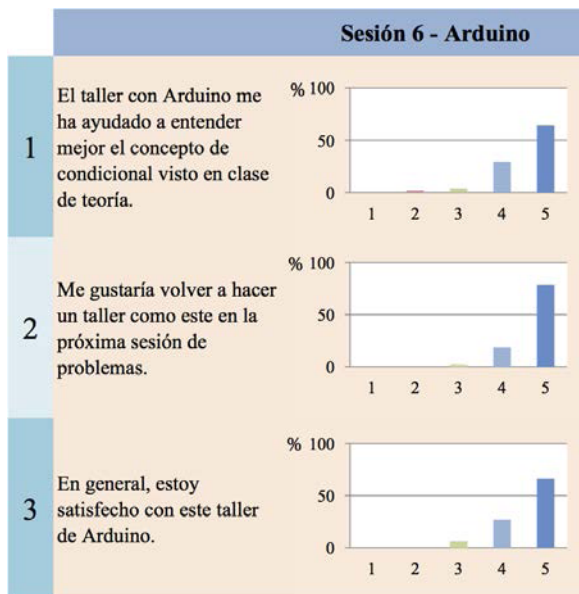
En este taller también se trabaja con más detención el uso de la función `pause`, necesaria en el bucle interno para ralentizar la ejecución y, de esta forma, permitir visualizar la secuencia de luces. Es interesante ver cómo, probando diferentes valores de tiempo y diferentes ubicaciones de la función (dentro del primer bucle, dentro del segundo bucle y fuera de los dos), los estudiantes consiguen entender de una manera visual el funcionamiento de los bucles anidados.

A partir de ahí, les animamos a hacer que el tiempo de espera sea variable durante la ejecución. Les planteamos que pueden, por ejemplo, acelerar la secuencia visual disminuyendo el tiempo de espera en cada ejecución. También les sugerimos que modifiquen el parámetro pasado a la función `disp` para mostrar en pantalla tanto el valor de los LEDs como el tiempo de espera. Por último, les explicamos el uso de la sentencia `break`. Les sugerimos que aceleren la secuencia visual hasta llegar a un umbral, a partir del cual la secuencia no se puede percibir debido a la alta velocidad de ejecución. Para ello deben utilizar una secuencia condicional dentro del bucle principal.

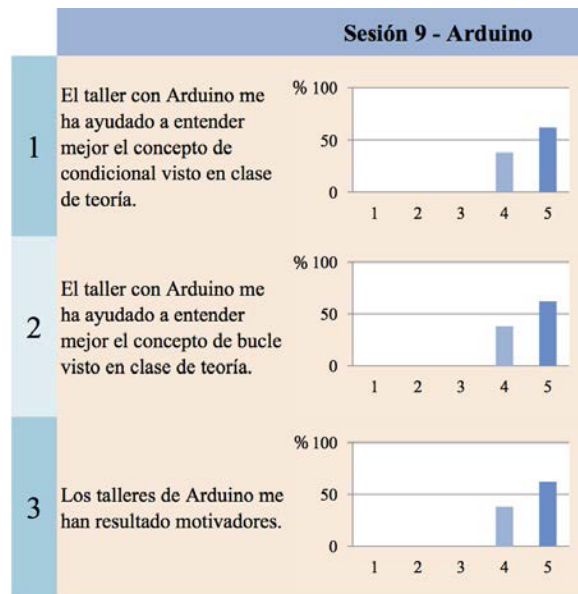
Con este ejercicio finalizamos el taller, quedando los profesores sorprendidos por el nivel de entusiasmo y motivación que muestran algunos alumnos.

6. Resultados

Para el estudio del impacto de las acciones realizadas se ha considerado principalmente la realización de encuestas a los estudiantes, lo que nos ha permitido evaluar el grado de satisfacción de los alumnos con los talleres Arduino y obtener feedback de los mismos. Las encuestas se pasaron de forma anónima a un total de 4 grupos de problemas, tres de ellos donde se realizaron los talleres (grupos de test), con un total de 48 estudiantes y un cuarto grupo donde se siguió la me-



Cuadro 1: Encuesta de satisfacción sobre los micro-talleres Arduino pasada a los alumnos durante la sesión 6 de problemas (grupos de test).



Cuadro 2: Encuesta de satisfacción sobre los micro-talleres Arduino pasada a los alumnos durante la sesión 9 de problemas (grupos de test).

todoología de los cursos anteriores (grupo de control), con un total de 12 estudiantes.

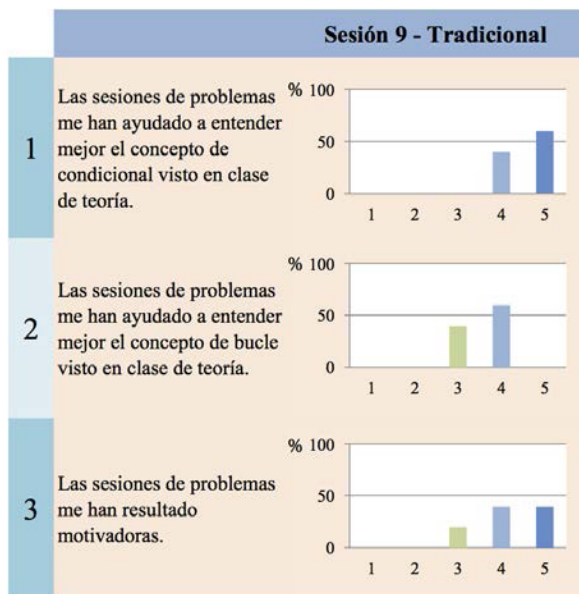
En los grupos de test se pasaron dos encuestas en dos momentos diferentes. La primera se pasó al acabar el primer taller de Arduino, durante la sesión 6 de problemas (ver Cuadro 1), donde el objetivo principal era practicar y afianzar el uso de condicionales. La segunda se pasó al acabar el último taller de Arduino, durante la sesión 9 de problemas (ver Cuadro 2), donde el objetivo principal era afianzar tanto el uso de condicionales como de estructuras repetitivas (bucles). Los resultados muestran, en general, respuestas positivas de los estudiantes. Las respuestas son aún más positivas en las segunda encuesta, después de la última sesión, en donde todas las respuestas están entre 4 y 5 en una escala Likert de cinco niveles. Como se puede observar, los alumnos consideran que la realización de los talleres les ha ayudado a entender los conceptos de bucles y condicionales, y también ha mejorado su motivación.

En lo que respecta al grupo de control, la encuesta se realizó al finalizar la sesión 9 de problemas (ver Cuadro 3). Los resultados indican que el uso del enfoque tradicional, basado en la resolución de ejercicios en papel, es menos efectiva para la comprensión de las estructuras de control que en el caso de la visión complementaria proporcionada por los micro-talleres. Por otra parte, la motivación de los alumnos al final de las sesiones no es tan alta como en los grupos de test.

7. Conclusiones

En este trabajo se ha presentado una propuesta de mejora metodológica cuyo objetivo es favorecer la comprensión y el aprendizaje, por parte de los estudiantes, de dos conceptos elementales en los que venimos encontrando serias dificultades de aprendizaje: estructuras repetitivas (bucles) y condicionales. La efectividad de esta propuesta, medida principalmente a través de encuestas, viene avalada por una valoración muy positiva por parte del alumnado, apuntando a un alto grado de motivación y a la consecución de los objetivos que nos habíamos planteado.

Pretendemos que estos talleres puedan ser utilizados en cualquier asignatura de introducción a la programación, tanto en titulaciones de informática como de ingenierías en general. El material para poder realizar los talleres se encuentra accesible en internet con la intención de que pueda ser utilizado, y si es el caso, mejorado y ampliado, por la comunidad educativa. Tanto el coste como el tiempo necesario para su puesta en marcha es razonablemente bajo. En el sitio web se pueden encontrar los boletines de problemas, los esquemas de montaje y vídeos ilustrando el resultado final que se desea conseguir. Pretendemos seguir mejorando y ampliando estos talleres en sucesivos cursos académicos y poder compartir comentarios, sugerencias y experiencias mediante el uso de la plataforma web.



Cuadro 3: Encuesta de satisfacción sobre las sesiones de problemas pasada a los alumnos del grupo de control (grupo en el que no se han realizado los microtalleres Arduino, habiéndose seguido la metodología tradicional) durante la sesión 9.

8. Agradecimientos

Este trabajo ha sido financiado por la Unitat de Suport Educatiu y el Vicerrectorado de Estudiantes, Ocupación e Innovación Educativa de la Universitat Jaume I, en el marco del Seminario Permanente de Innovación Educativa SPIE 2962/14 y el proyecto de innovación educativa asociado y el Grupo de Innovación Educativa GIE 2935/14, y por el plan estratégico 2014-15 del Departamento de Ingeniería y Ciencia de los Computadores de la Universitat Jaume I.

Referencias

- [1] J. Sales y G. Recatalá. Mejora del rendimiento en las sesiones de laboratorio de la asignatura Informática para ingenierías. En Simposio-Taller XIX Jornadas de Enseñanza Universitaria de la Informática, Jenui 2013, páginas 53 – 60, Castellón, España, Julio 2013.
- [2] M. Fernández, C. Hernández, G. Recatalá y J. Sales. Desarrollo de actividades de Autoaprendizaje y Cambio de Metodología Docente en asignaturas no presenciales o presenciales basadas en problemas y proyectos. En II Congreso Internacional sobre Aprendizaje, Innovación y Competitividad (CINAIC 2013), páginas 548 – 553, Madrid, España, Noviembre 2013.
- [3] G. Recatalá, J. Sales, M. Fernández, J.V. Martí y C. Hernández. Estrategia de resolución de problemas y trabajo en grupo para la mejora de la participación activa. En Simposio-Taller de las XX Jornadas de Enseñanza Universitaria de la Informática, Jenui 2014, páginas 27 – 34, Oviedo, Julio 2014.
- [4] M.C. Aranda, A.J. Fernández, J. Galindo y M. Trella. Valoración del Marco Docente de la Informática en la Ingeniería Técnica Industrial: Propuesta de una Nueva Metodología. En Actas de las VII Jornadas de Enseñanza Universitaria de la Informática, Jenui 2001, Palma de Mallorca, Julio 2001.
- [5] A.B. Moreno, J.J. Pantrigo y R. Peña. Propuesta para la enseñanza de Informática en titulaciones de Ingeniería Química. En Actas de las IX Jornadas de Enseñanza Universitaria de la Informática, Jenui 2003, páginas 199 – 206, Cádiz, Julio 2003.
- [6] Julio A. Hurtado, César A. Collazos, Samith Tatiana Cruz, Oscar Eduardo Rojas. Child Programming: Una estrategia de aprendizaje y construcción de software basada en la lúdica, la colaboración y la agilidad. Revista RUTIC, Vol.1 No.1, 2012.
- [7] D. May, K. Lensing y otros. What students use - Results of a survey on Media Usage among Engineering Students. En proceedings of 2014 Frontiers in Education Conference, FIE 2014, páginas 92 – 97, Madrid, España, Octubre 2014.
- [8] D. Deb, M.M. Fuad, W. Farag. Developing interactive classroom exercises for use with mobile devices to enhance class engagement and problem-solving skills. En proceedings of 2014 Frontiers in Education Conference, FIE 2014, páginas 343 – 346, Madrid, España, Octubre 2014.
- [9] J. Pastor, P.A. Revenga, M. Salazar, A. Sánchez, C. de la Rubia y otros. Depecabot. A hands on Workshops to teach robotics and improve students electronics knowledge. En proceedings of 2014 Frontiers in Education Conference, FIE 2014, páginas 347 – 352, Madrid, España, Octubre 2014.
- [10] M.A. Rubio, C. Mañoso, R. Romero, A.P. de Madrid. Uso de las plataformas LEGO y Arduino en la enseñanza de la programación. En Actas de las XX Jornadas de Enseñanza Universitaria de la Informática, Jenui 2014, páginas 419 – 426, Oviedo, Julio 2014.
- [11] D. Calinoiu, R. Ionel, M. Lascu, A. Cioabla. Arduino and LabVIEW in educational remote monitoring applications. En proceedings of 2014 Frontiers in Education Conference, FIE 2014, páginas 245 – 249, Madrid, España, Octubre 2014.